Carnegie Mellon University

Information and Communication Technologies Institute
Carnegie Mellon | PORTUGAL
AN INTERNATIONAL PARTNERSHIP

TÉCNICO LISBOA

Personalized Assistance for Dressing Users

Steven D. Klee[1], Beatriz Quintino Ferreira[2], Rui Silva[3], João Paulo Costeira[2], Francisco S. Melo[3], and Manuela Veloso[1]

[1] Computer Science Department, Carnegie Mellon University, PA, USA
[2] ISR and Instituto Superior Técnico, Lisboa, Portugal
[3] INESC-ID and Instituto Superior Técnico, Lisboa, Portugal

ICSR 2015
INTERNATIONAL CONFERENCE ON SOCIAL ROBOTICS

Introduction

- New paradigm: robots and humans operating in the same workspace
- Human presence introduces uncertainty
- Designing robotic systems to deliver personalized assistance to different users is challenging

Problem: Provide personalized assistance to help dress a person with a manipulator

Main contributions

Key concept: collaborative turn-taking

Main contributions

Key concept: collaborative turn-taking

An approach to human-robot interaction that

- **plans** over requests the robot can ask the user,
- generates **expectations** based on those requests,
- **monitors** the expectations, and
- **learns** a constraint-model for the user when they behave differently

Approach to Personalized Dressing

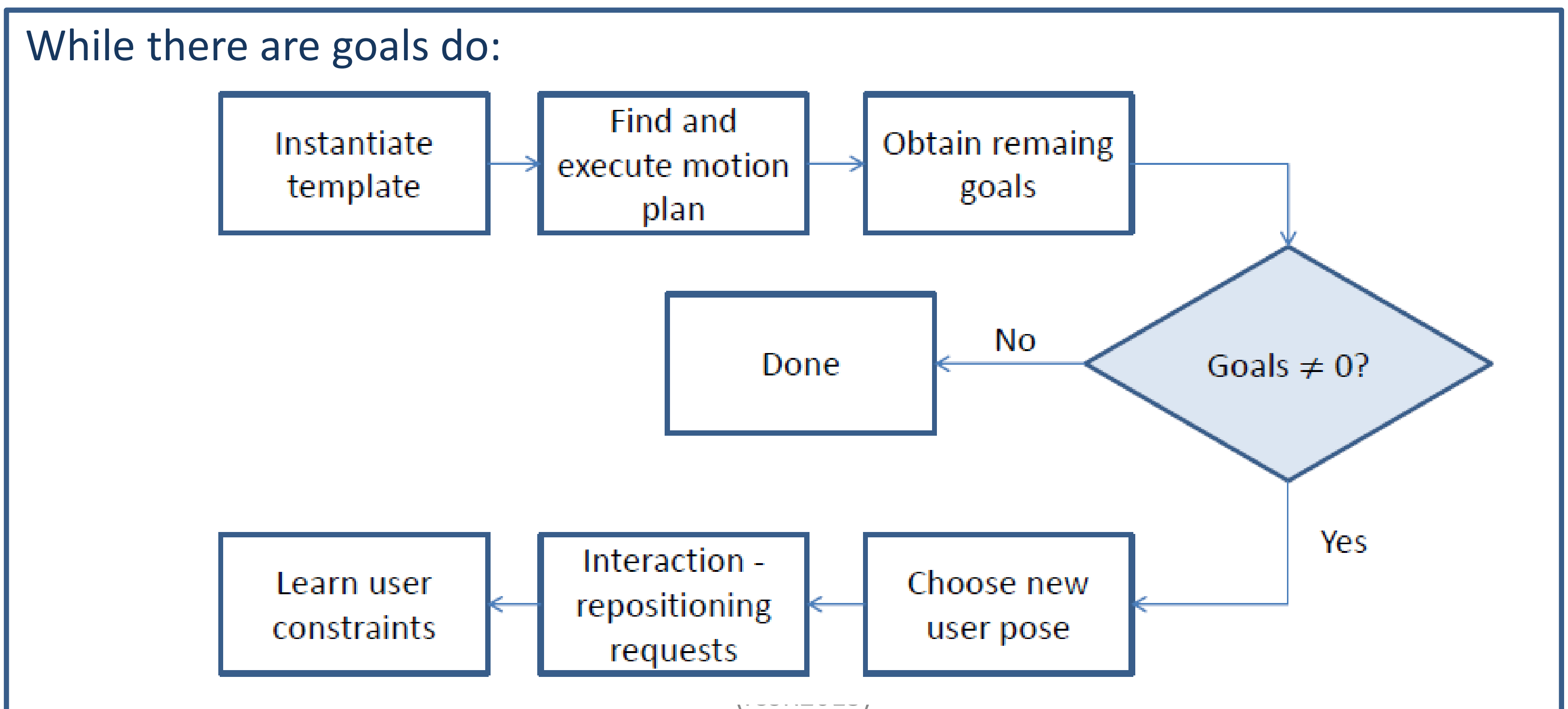
- **Our aim:** enable a manipulator to aid users in dressing tasks by providing personalized assistance
- **Motivating example:** help a user to put on a hat

Fig. 1: Baxter manipulator ready to help dress a user with two different hats

Approach to Personalized Dressing

- Each dressing task is a *template*
- Each *template* is a sequence of *goal* poses
- A template is instantiated with the current user position and orientation

Execution of Personalized Human-Robot Motion Interactions:



Approach to Personalized Dressing

Vision-Based User Tracking

Functions:

- Monitors user to pause motion execution if user moves
- Verifies if user is complying with repositioning requests

Person modeled as a set of connected body parts B

For each body part $\mathrm{b} \in B$, we compute

center-of-mass location (x, y, z)

orientation (q_x, q_y, q_z, q_w)

to instantiate the goals for the motion planner

Approach to Personalized Dressing

Dressing Task as Template Goals

Dressing task represented as a sequence of manipulator pose goals w.r.t. the user

E.g. Putting on a hat: one desired pose goal

Putting on a backpack: several desired pose goals

$$T = \langle P_1, P_2, \ldots, P_n \rangle \qquad P_i = \langle x, y, z, q_x, q_y, q_z, q_w, TOL \rangle$$

Approach to Personalized Dressing

Motion Planning

- **Sparse-interaction** approach
- **Safety**:
  - manipulator only moves when the user is stopped
  - if user begins moving robot halts and re-plans once user stops

- Sampling-based motion planner: RRT-Connect
- Infeasible poses: manipulator determines a sequence of user repositioning requests to a new location

Approach to Personalized Dressing

User Aware Pose Selection and Assistance Planning

- Robot has a knowledge-base of user constraints
- Future poses are sampled from the set of known feasible manipulator poses that satisfy all *active* user constraints
- Constraint c on body part $b \in B$ is represented as:

$$c = \langle b, ineq, conf \rangle$$

- Repositioning requests are asked through *motion actions*

E.g. *forwards(x), left(x), up(x), turn_right(Θ)*

Approach to Personalized Dressing

Learning and Refining User Constraints

- Robot learns and infers constraints in two situations:
  1. User stops before reaching expected pose during repositioning request

     Constraint c_b is generated for body part $b \in B$:

     $$c_b = \langle b, b.x > vision.b.x, conf \rangle$$

  2. User enters a space the robot believed to be constrained

     Original constraint c_b is relaxed
- In any case, the robot updates its confidence in the constraint

Evaluation

- Baxter Manipulator
  - Force compliant motion
- Microsoft Kinect depth sensor
  - OpenNI-Tracker

Task: Put on a hat on the user’s head
(users ablated themselves to simulated physical limitations)

Evaluation

Increasing Constraint Complexity

Hypothesis: Users with more complex constraints take longer to dress

↓

motivates learning personalized constraint models

- Tests in the real world with two users: taller person (1.83m) and shorter person (1.62m)
- 2 simulated constraints

Evaluation

Increasing Constraint Complexity

	User in feasible pose	User repositions to a feasible pose	User has 1 constraint, robot starts by user-infeasible pose	User has 2 constraint, robot starts by user-infeasible pose
Execution time (s)	11.6 ± 1.8	32.0 ± 12.3	53.0	78.0
Number of Interactions	0	2.1	3.3	3.5

Table 1: Average execution times and number of interactions for the 4 considered cases, over 40 trials

The two users took different times to respond to robot requests

Evaluation

Dressing task execution for a user constrained to a chair

a) User is initially at an unreachable pose (user too far away, motion planner cannot solve goal)

Evaluation

Dressing task execution for a user constrained to a chair

a) User is initially at an unreachable pose (user too far away, motion planner cannot solve goal)

b) Robot selects new pose believed to be feasible for the user and asks them to move forward

Evaluation

Dressing task execution for a user constrained to a chair

a) User is initially at an unreachable pose (user too far away, motion planner cannot solve goal)

b) Robot selects new pose believed to be feasible for the user and asks them to move forward

c) Template goal is re-instatiated with new feasible position. Robot solves the goal and successfully places hat on the user's head

Evaluation

Dressing task execution for a user constrained to a chair

a) User is initially at an unreachable pose (user too far away, motion planner cannot solve goal)

b) Robot selects new pose believed to be feasible for the user and asks them to move forward

c) Template goal is re-instatiated with new feasible position. Robot solves the goal and successfully places hat on the user's head

40 trials

- Large majority (37) successful
- Occasional errors due to positions on the extremes of reachability, or occlusions in the vision module

Evaluation

Learning User Models

- Robot was taught a constraint model (with 2 constraints) for each user

	Shorter user	Taller user
Execution time (s)	34.2 ± 29.4	36.5 ± 8.5
Number of Interactions	2.6	2.6

Table 2: Average execution times and number of interactions for the two users, over additional 5 trials resorting to the knowledge-base formed by previous interactions

Evaluation

Learning User Models

	Shorter user	Taller user
Execution time (s)	34.2 ± 29.4	36.5 ± 8.5
Number of Interactions	2.6	2.6

Table 2: Average execution times and number of interactions for the two users, over additional 5 trials resorting to the knowledge-base formed by previous interactions

As seen before:

	User repositions to a feasible pose
Execution time (s)	32.0 ± 12.3
Number of Interactions	2.1

Table 3: Average execution time and number of interactions for the case in which user has no limitations and repositions after robot request

Evaluation

Learning User Models

	Shorter user	Taller user
Execution time (s)	34.2 ± 29.4	36.5 ± 8.5
Number of Interactions	2.6	2.6

Table 2: Average execution times and number of interactions for the two users, over additional 5 trials resorting to the knowledge-base formed by previous interactions

Learning personalized constraint models can markedly optimize task execution

Conclusion

- Approach for a manipulator to help dress users, as a sequence of pose goals
- The robot asks the users to reposition themselves when the goal is infeasible
- Repositioning poses chosen by sampling points in the robot configuration space that satisfy user-constraint model
- Approach demonstrated on a Baxter manipulator
- Modeling user constraints makes dressing task more efficient (time similar to case with no constraints)